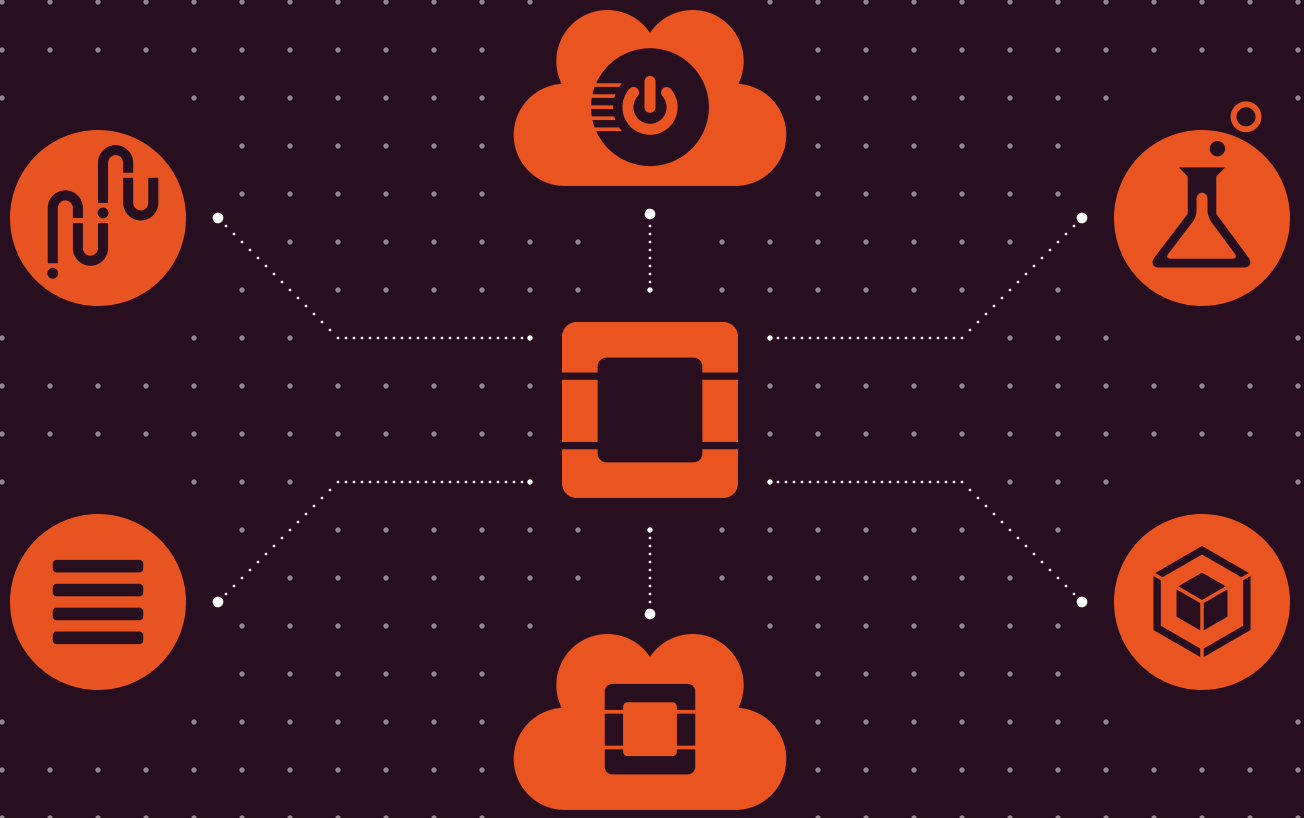
eBook

ubuntu

OpenStack made easy



What you will learn

We're in a phase change from traditional, monolithic scale-up software to multi-host, scale-out microservices. Welcome to the age of 'big software'.

Big software demands that infrastructure and operations personnel approach the challenge of deployment, integration and operations from a different perspective. This eBook explains how we used to do things, why that is no longer an economically viable approach, and and what can be done to achieve technical scalability without the large economic overhead of traditional approaches to modern software.

This eBook will also help you gain a deeper understanding of why there is a perceived complexity to the installation and operations of OpenStack-based clouds. You will learn that this perceived complexity does not originate from the software itself, but rather the use of outdated tools and methodologies used to deploy them.

This eBook will explain how Canonical and Ubuntu are uniquely positioned to facilitate the needs of modern, scalable, repeatable implementations of cloud infrastructure based on OpenStack.

Many of the approaches that we discuss in this eBook are also useful when addressing other big software challenges in areas such as scale out applications and workloads, big data and machine learning.

Contents

What is OpenStack?

General overview

OpenStack is a collection of open source software projects designed to work together to form the basis of a cloud. Primarily, it is used for private cloud implementations, but it can be just as applicable for cloud service providers to build public cloud resources. It's important to understand that OpenStack is not a single product, but rather a group of projects.

Modular

From its inception, OpenStack was designed to be modular and to be integrated with additional tools and plugins via APIs. You could choose to use any single project from OpenStack to accomplish a particular task, or several of them, to build out a more complete cloud.

Canonical integrates the projects, along with additional components, into a fully fledged enterprise Cloud Platform known as Canonical OpenStack.

Core projects and more

The core projects of OpenStack consist of Nova (compute), Neutron (networking), Horizon (dashboard), Swift (object storage), Glance (image storage), and Keystone (identity).

Beyond the core projects, there are additional solutions and tools in the industry to enhance the deployment, integration and daily operation of an OpenStack cloud.

OpenStack challenges

Hardware configuration

Most organisations still manage some hardware. After racking and connecting it, initial configuration must be done. Some use vendor tools, some write proprietary scripts, others leverage ever-growing teams of people. Some use a combination of all of these approaches and more.

The issue with these approaches is economic scalability. If you change hardware configuration in any way, you need to pay to add/modify an ever-growing collection of scripts. If you change hardware vendor, you need to add, configure and maintain a new tool, while maintaining all previous hardware management tools. If you add more servers, you have to hire more people. None of this scales with cloud economics.

Hardware integration

Beyond the initial configuration, integration must happen. Network services must be set up and maintained, including DHCP or static IP address pools for the host NICs, DNS entries, VLANs, etc. Again, these integration tasks can be accomplished with scripts, vendor tools or personnel, but the same potential issues arise as with configuration.

OpenStack intallation

Another major obstacle to OpenStack success is the initial installation. The aforementioned scripting approach is common, as are growing teams of expensive personnel.

There are also OpenStack projects to perform installation, but they are often vendor-driven, not neutral and lack feature completeness. Organisations that try to use them often find themselves doing significant, ongoing development work to make the project useful.

Additional challenges

On-going challenges all lend to increasing cost and decreasing economic scalability. Additional considerations include:

- Upgrades
- Rebuilding
- New clouds
- Repeatable best practices
- Scaling out
- Reducing cost of consultants

A scalable, practical approach

A better approach, an easier approach, are vendor, hardware and platform neutral tools. Tools that include APIs for automation of not just software, but your datacenter, as well. Tools with graphical interfaces, designed with scalable cloud economics in mind.

Putting the intelligence of installation and integration complexity directly into the tools themselves is how you make OpenStack easy and achieve economic scalability.

OpenStack installation and integration challenges are best solved by a thoughtful approach, using technologies designed for modern clouds. Legacy scripting technologies might work now, but likely won't scale as your cloud's needs change and grow. The same goes for personnel.

This eBook will go into detail about the approach and tools that make OpenStack easy.

Who is Canonical?

CANONICAL

The company

Canonical is the company behind Ubuntu, the underlying Linux server platform for 65% share of workloads on public clouds and 74% share of OpenStack deployments. We are the leading platform for OpenStack with 55% of all production OpenStack clouds built with Ubuntu.*

A founding member of the OpenStack Foundation, Canonical also has a long history of interoperability testing between OpenStack, Ubuntu and partner technologies. Its OpenStack Interoperability Lab (OIL) currently tests over 3,500 combinations per month.

Market focus

Canonical is focused on cloud scalability, economically, and technologically. That means focusing on density with containers, operational efficiency with application modeling and financial scalability with cloud-optimized pricing.

We have proven success supporting large scale cloud customers in production, with some examples given on the Canonical OpenStack in production page of this eBook.

Choice

Solutions from Canonical are hardware agnostic, from platform to processor architecture and public cloud options. We recognize that modern organisations require flexibility and choice. The tools discussed in this eBook that enable ease of use and decreased operational costs are designed to work across all platforms and major clouds, not just select partners.

*Source: OpenStack User Survey 2016

OpenStack Interoperability Lab (OIL)

Proven integration testing

Canonical has a long history of interoperability testing between OpenStack and Ubuntu.

The Openstack Interoperability Lab (OIL) is the world's largest Openstack interoperability and integration test lab. It is operated by Canonical with over 35 major industry hardware and software partners participating. Each month we create and test over 3,500 cloud combinations in the OIL lab. We could not do this without the solutions described in this ebook.

Sophisticated testing and integration processes

Our process tests current and future developments of OpenStack against current and future versions of Ubuntu Server and Ubuntu Server LTS.

As our ecosystem has grown, we've expanded it to include a wide array of guest operating systems, hypervisors, storage technologies, networking technologies and software-defined networking (SDN) stacks.

Why OIL makes OpenStack easier

OIL ensures the best possible user experience when standing up your Canonical OpenStack cloud and maintaining it.

By testing up to 500,000 test cases per month you can run your Canonical OpenStack cloud and technologies from our partner eco-system with greater ease and confidence.

Find OIL partners

Canonical OpenStack in production

Almost all OpenStack projects are developed, built and tested on Ubuntu. So it's no surprise that Ubuntu OpenStack is in production at organizations of all sizes worldwide. Over half of all production OpenStack clouds are built with Ubuntu.

To give you an idea of what organisations are doing with Canonical OpenStack, we've highlighted a few here:

- Deutsche Telekom
- Walmart
- Tele2
- Sky
- AT&T

German telco Deutsche Telekom (DT) uses Ubuntu OpenStack as the foundation of a next-generation network functions virtualisation (NFV) infrastructure.

Deutsche Telekom leverages Canonical's tool chain even further, using Juju as a generic Virtualised Network Functions (VNF) manager. In this case, Juju is used to model and deploy both OpenStack, as well as the critical workloads running within the Ubuntu OpenStack environment.

"When I started working with OpenStack it took three months to install. But with the help of Juju it takes three days"

Robert Schwegler
Deutsche Telekom AG

Walmart

Walmart, an American multinational retail corporation, uses Ubuntu OpenStack as the foundation of their private cloud.

One of the key factors of scalability is economics. Here, the economic scalability of Canonical OpenStack cannot be overlooked. While the technology is certainly designed to scale, it's just as critical that the methodologies for deployment and billing are also designed to scale.

"[Canonical] OpenStack met all the performance and functional metrics we set ourselves… It is now the defacto standard and we can adapt it to our needs"

Amandeep Singh Juneja
Walmart

TELE2

A major European telecommunications operator, with about 14 million customers in 9 countries, Tele2 has also built an NFV infrastructure on Ubuntu OpenStack.

They have opted for a BootStack cloud; a fully managed Ubuntu OpenStack offer from Canonical

BootStack dramatically reduces the time it takes to bring OpenStack into production, and allows Tele2 to focus their skilled resources on telecoms solutions, and not having to learn and update their skills to the fast-paced changes of OpenStack.

And many more…

NTT, Sky Group, AT&T, Ebay, Samsung and many other organisations all represent customers that have elected to build clouds on Canonical OpenStack.

Scalable technology, scalable economics, ease-of-use and reduced time to solution are the primary reasons that so many organisations choose Canonical OpenStack.

"When we started our private cloud initiative we were looking for a sustainable cost base that makes it effective and viable at scale…

we needed a platform that was robust and brings innovation. Canonical OpenStack helps us meet and realise those because of the broad experience Canonical brings"

Will Westwick
Sky Group

"We're reinventing how we scale by becoming simpler and modular, similar to how applications have evolved in cloud data centers.

Open source and OpenStack innovations represent a unique opportunity to meet these requirements and Canonical's cloud and open source expertise make them a good choice for AT&T"

Toby Ford
AT&T

More than one cloud

Value of repeatable operations

When building OpenStack clouds it's important to understand the need for repeatable operations.

One of the common conceptions of building and operating a cloud is that you do it once and it's done. There is a tendency to put tremendous time and effort into designing both the physical and software infrastructure for what is to be a static production cloud. Often there is little thought put into rebuilding it, modifying it, or doing it many times over.

The reality of modern clouds is that there is no static production cloud that is never upgraded or expanded to more than one cloud or rebuilt as part of a rolling upgrade.

Also, there is no one size fits all cloud.

Successful early cloud adopters have come to realize that remote locations may have each their own small cloud infrastructure. For scalability and redundancy, even within a single datacenter, they will end up building many, even dozens, of clouds.

Telcos, media and broadcast companies and enterprise organisations distribute operations globally, with potentially thousands of smaller, off-site operations centers. All need their own cloud to support localised and scalable infrastructure.

Even smaller organisations build development, test, staging and production clouds.

Everyone needs to do these builds consistently, in a repeatable fashion, many times.

The perception of difficulty with big software like OpenStack

There's a perception that OpenStack is difficult to install and maintain without expert knowledge. This perception largely stems from a flawed approach. OpenStack is 'big software' which means it has so many distributed components that no single person can understand all of them with expert knowledge. Yet, organisations are still looking for individuals, or teams of people who do.

The larger the cloud, the more solutions run on it, the more people they think they need. This approach is not scalable economically or technically.

A modern look at the OpenStack perception of difficulty reveals that the best practices for installation, integration and operations should be distilled into the software itself. The knowledge should be crowdsourced and saved in bundles that encapsulate all of the operational expertise of the leading industry experts so that it can be easily and repeatably deployed. That is what Canonical has done that has made Ubuntu OpenStack so successful.

In the pages ahead we will show how this practice has been adopted for both hardware, with MAAS and conjure-up, as well as software, with Juju.

The challenge of big software

In his keynote at the OpenStack Summit Austin 2016, Mark Shuttleworth, Executive Chairman of Canonical and lead of the Ubuntu project, demonstrated how big software like OpenStack can be fast, reliable & economic.

Watch Video

Operational intelligence at scale

In order to scale, operational intelligence must no longer be a function of number of skilled operators, but rather a function of the right tools designed to focus on the right issues. This is where Canonical's unique toolset makes Canonical OpenStack relatively easy compared to other offerings.

Tools built specifically for big software like OpenStack are the only way to achieve cloud economics in a private cloud. Adding personnel won't scale as your cloud grows, and using traditional scripting technologies requires too many, and too frequent of updates within a growing, dynamic cloud.

In the next section, we introduce MAAS, to manage bare metal hardware, Juju, to manage application design and deployment, and conjure-up, to completely automate the configuration, architecture choices and deployment of a Canonical OpenStack cloud.

Additional tools and solutions are introduced, as well. Conjure-up can deploy single-node or multi-node OpenStack test/dev environments through an intuitive interface and setup test OpenStack deployments in VMware or even public cloud.

Since containers are vital to system density and return on investment, we will also discuss how LXD, the pure container hypervisor, and Fan networking, play essential roles in solving for server and IP network density.

MAAS – the smartest way to handle bare metal

Why MAAS?

Hardware must still be installed in a datacentre. The key to economic efficiency is to touch it as few times as possible. Installing and operating a bare metal OS at scale won't work if done by hand or custom scripts for every machine type.

MAAS stands for 'Metal as a Service'. MAAS delivers the fastest OS installation times on bare metal in the industry thanks to its optimised image-based installer.

Hardware configuration

With MAAS, you only touch the power button once. During the initial startup of a new server, MAAS indexes it, provisions it, and makes it cloud ready. A catalog is maintained of not only the servers, but also the inventory of devices available in them. This is a key aspect of future provision automation by Autopilot.

Ongoing infrastructure operations

Beyond initial configuration, MAAS also handles ongoing physical IP and DNS management. A "lights out" datacentre, with a near-zero need for hands-on operations, is realized with MAAS.

Accessible

MAAS provides a REST API, Web-based interface and command line interface. It is designed with automation and hardware-at-scale in mind. Devops can even leverage it for bare metal workload management.

Integration

Since there's an API, as well as a CLI, automation tools like Juju, Chef, Puppet, SALT, Ansible, and more, are all easily integrated with MAAS. That means legacy, scripted automation, like Puppet and Chef, are easily integrated, whilst modern modeling tools, like Juju, can naturally rely on MAAS for hardware information.

Learn more about MAAS at maas.io

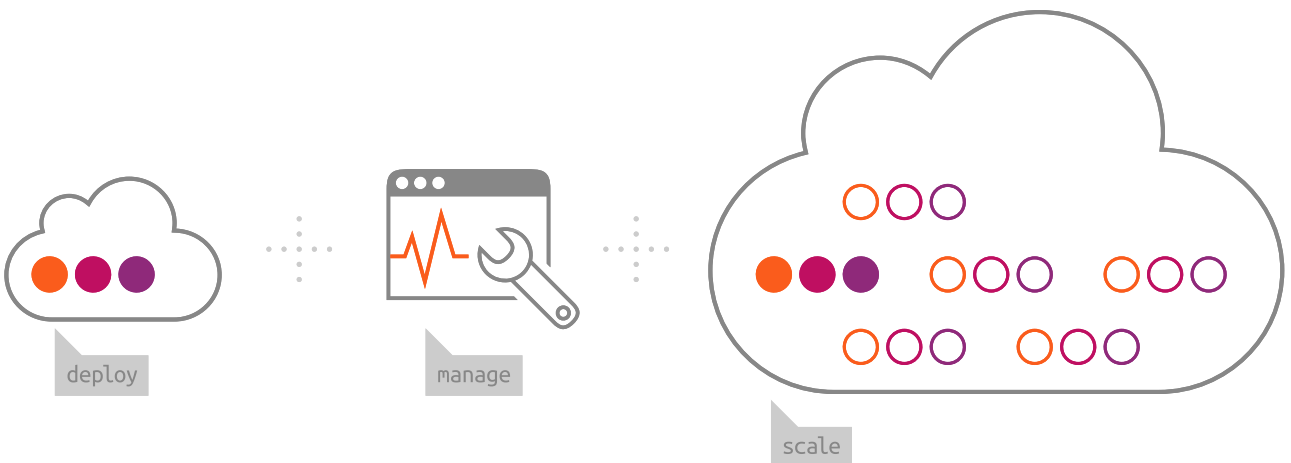
Juju – model-driven operations for hybrid cloud services

Why Juju?

It's challenging to model, deploy, manage, monitor and scale out complex services in public or private clouds. As an application and service modelling tool, Juju enables you to quickly design, configure, deploy and manage both legacy and cloud ready applications.

Juju has been designed with the needs of big software in mind. That is why it is not only leveraged by conjure-up for OpenStack installation and updates, but it can also be used to deploy any scalable application. All of this is possible from a web interface or with a few commands.

Juju can be used to deploy hundreds of preconfigured services, OpenStack, or your own application to any public or private cloud.



Web UI and Command Line Interface

Juju's user interface can be used with or without the command line interface. It provides a drag-and-drop ability to deploy individual software or complex bundles of software, like Hadoop, or Ceph, performing all the integration between the associated components for you.

You have a graphical way to observe a deployment and modify it, save it, export it. All of this can be done at command line, as well.

Charms encapsulate best practices

Juju is the key to repeatable operations. Juju uses Charms that encapsulate operational intelligence into the software itself that is being deployed. The best practices of the best engineers are encapsulated in Charms. With Juju, you don't need an expert in every OpenStack project, and an expert in every big software application, like Hadoop, in order to achieve operational excellence. All you need is an understanding of the application(s) once it's been deployed using the crowdsourced operational excellence in Juju's Charms and bundles.

Learn more about Juju at jujucharms.com

Conjure-up – the fastest way to build an on-premise cloud

Why Conjure-up?

Many organisations find building a production OpenStack environment challenging and are prepared to invest heavily in cloud experts to achieve operational excellence. Just like Juju and Charms, conjure-up encapsulates this operational excellence.

Conjure-up is also able to deploy other big software applications such as Canonical Kubernetes so has become the goto application for deploying distributed service based software applications.

A decision engine for your cloud

While conjure-up allows the user to manually determine hardware allocations, it's generally best left to the decision engine within it. The underlying infrastructure is modelled by MAAS and shared with conjure-up. Availability zones are automatically created for you.

Once the cloud is deployed operators use the underlying Juju to manage the environment and easily bind into services such as log management and monitoring.

Build the cloud architecture that works for you

The reference architecture that conjure-up will automatically design for you will accomplish maximum utilisation of the resources given to the cloud.

Conjure-up is not overly prescriptive though as our experience shows that all cloud deployments will go through several iterations before settling on an architecture that works. Conjure-up enables this by offering full flexibility of service placement across the available physical infrastructure so that you can get the cloud that works for you.

Learn more about conjure-up at conjure-up.io

Containers in OpenStack

Why containers?

Containers have many benefits. But there are two things they do extremely effectively. One, package applications for easier distribution. That's an application container like Docker. The other is to run both traditional and cloud-native workloads at bare metal speed. That's a machine container, like LXD. Application containers can even run inside machine containers, to potentially take full advantage of both technologies.

Why now?

As more workloads move to clouds like OpenStack, the economies of scale are affected not only by the right tools and the right approach, but the right workload density as well. We run more workloads on a given server than ever before. The fewer resources a given workload needs, the greater the return on investment for a cloud operator, public or private.

OpenStack containers made easy

While container technology is extremely compelling, there can be some difficulties in integration, operation and deployment. With the nova-lxd technology in Ubuntu 16.04, a pure container OpenStack deployment is easily achieved. Nova-lxd provides native integration for OpenStack with LXD machine containers. That means that no extra management software is needed to deploy both traditional virtual machines as well as modern machine containers from a native OpenStack API or Horizon dashboard.

LXD – the pure container hypervisor

LXD, the pure container hypervisor, is the key to delivering the world's fastest OpenStack, as demonstrated at OpenStack Summit in Austin, TX. It achieves the lowest latency and bare metal performance.

LXD helps enable a hyperconverged Canonical OpenStack. It deploys in minutes. Instances that run on top of OpenStack perform at bare metal speed. Dozens of LXD instances can be launched within that OpenStack cloud in a matter of seconds.

When using LXD, an entire OpenStack environment can be snapshotted in about 2 seconds.

Operational efficiency is furthered by the ability to live migrate services from one physical host to another, just like legacy hypervisors, but with a pure container hypervisor.

Upgrading a host's LXD containers is as simple as upgrading the underlying OS (Ubuntu), migrating services off and back.

You can even run LXD containers inside other LXD containers; all at bare metal speed, with no performance degradation. Traditional virtual machines must run on bare metal and cannot be run practically inside other VMs.

There are prebuilt LXD images for running CentOS, Debian, OpenSUSE, Fedora, and other Linux operating systems.

Security is implicit with mandatory access controls from Apparmor profiles. LXD pure containers are as secure as Linux itself.

LXD can run virtually any Linux distribution as a guest operating system. It doesn't require special virtualisation hardware. It even allows you to deploy all of OpenStack inside another cloud, like on Amazon, for example.

Learn more about LXD at ubuntu.com/lxd

Fan networking – network addressing for containers

Why Fan networking?

The density and performance of both machine containers (LXD) and application containers (like Docker) are extremely compelling for modern cloud economics. But, their operation, specifically when it comes to network addressing, can be problematic.

In the case of application containers, each application, or binary, requires a unique IP address. With potentially hundreds of containers on any individual server, IP addresses are quickly depleted.

While there are network addressing workarounds available, like port-forwarding, they just shift an administrative burden from one technology, to another. In this case, it is now port management.

Network address expansion with Fan

A much more elegant solution is Fan networking. The Fan is an address expansion technology that maps a smaller, physical address space, into a larger address space on a given host. It uses technologies built into the Linux kernel to achieve near-zero loss of network performance while providing unique IP's to hundreds or even thousands of container guests.

Fan networking is another example of how Canonical is taking a thoughtful, meticulous approach to big software and OpenStack deployments. Instead of shifting the burden of a given issue from one administrative domain to another, the issue is addressed at its core, using best practices and partnership with the open source software community.

Learn more about Fan networking on on the Ubuntu Insights blog

Multi-node OpenStack deployment on your laptop

Why install on a single node?

In the past developing and testing software in an OpenStack environment has meant using OpenStack installers like DevStack. Whilst convenient, DevStack's monolithic architecture can't emulate a multi-node cloud environment.

Conjure-up is a command line tool exclusive to Ubuntu 16.04 that enables developers to easily deploy real-world OpenStack on a single laptop using LXD containers.

Multi-node OpenStack using LXD

Since LXD containers are like virtual machines, each OpenStack control node service is independent, even on a single physical machine.

Multiple physical machines are also an option, to further mimic production environments in development and test, without the complications of an entire datacenter of hardware.

Get started with conjure-up

Using conjure-up is easy if you already have Ubuntu 16.04. It's as quick as –

```
$ sudo apt install conjure-up
$ conjure-up
```

Learn more about conjure-up at conjure-up.io

ZFS and software defined storage

ZFS makes better containers

ZFS accelerates LXD on Linux. Specifically, it provides:

- Copy-on-write
- snapshot backups
- continuous integrity checking
- auto repairs
- efficient compression
- deduplication

All of these features improve the management and density of containers.

Container characteristics

Critical aspects of a successful container hypervisor are:

- Density
- Latency
- Performance
- Fast, secure, efficient

The features of ZFS make innovative and superior pure container technologies like LXD even better.

All clouds store data

Ubuntu Advantage Storage provides support for a number of software defined storage solutions, all priced at cloud scale. Ceph object storage is a popular technology that is readily available within Ubuntu OpenStack and provides massive scale-out storage for organisations of all sizes.

Another unique advantage to Canonical OpenStack and Ubuntu Advantage Storage is the CephDash, which provides real-time data analytics of Ceph deployments.

Learn more about Ubuntu cloud storage at ubuntu.com/storage

BootStack – your managed cloud

Why BootStack?

Even with the most advanced tools and the best teams, it can be a lot easier to get started with some help from the experts that build thousands of Canonical OpenStack clouds every month.

BootStack (which stands for 'Build', 'Operate' and 'Optionally Transfer') is a managed service offering that gets you an OpenStack private cloud in a matter of weeks, instead of months.

Build, Operate and Optionally Transfer

Canonical's cloud experts will design and build an Canonical OpenStack cloud to your specifications. The hardware can be hosted at your datacenter or a 3rd-party provider.

When you feel comfortable managing your OpenStack environment, there is an optional transfer of administrative ownership over to your internal team.

Another option is BootStack Direct, which includes training as Canonical builds out your OpenStack cloud. Once the cloud is operational, administration of the cloud is directly transferred to your team.

With BootStack and BootStack Direct, it has never been easier to instantiate an Ubuntu OpenStack cloud. Regardless of the BootStack offer you may choose, the cloud will still be built with the aforementioned toolset in this eBook, to best practices and reference architecture standards, as defined by those tools.

Learn more about BootStack at ubuntu.com/bootstack

Conclusion

OpenStack may not be easy, but it doesn't have to be difficult. The ease of OpenStack is in the approach. Big software can't be tackled with legacy tools and old fashioned thinking.

With the right tools, OpenStack can be easy, and it can reap financial rewards for your organisation:

MAAS is the smartest way to handle bare metal

Juju enables easy model-driven operations for hybrid cloud services

conjure-up is the fastest way to build an OpenStack cloud for production and also the simplest way for developers to build a multi-node OpenStack deployment on their laptop

LXD pure containers hypervisor, ZFS and Fan networking let you run traditional and cloud-native workloads at bare metal speed

BootStack is the easiest way to stand up your production cloud and have it managed by the world's leading OpenStack experts

If you're excited to hear more and talk to us directly, you can reach us on at:
Contact Us

To learn more about a managed solution for big data, download the datasheet:
BootStack Your Big Data Cloud

If you want to start trying things out immediately, we highly encourage you to visit:
jujucharms.com

Enjoyed this eBook?
You might also be interested in…

CIO’s guide to SDN, NFV & VNF

Why is the transition happening and why is it important?

Networking and communications standards and methodologies are undergoing the greatest transition since the migration from analogue to digital. The shift is from function-specific, proprietary devices to software-enabled commodity hardware.

Read this eBook to:

- Familiarise yourself with the three most popular terminologies today – SDN, NFV, and VNF
- Learn why the transition is happening
- Understand why it’s important for anyone responsible for a network to understand and embrace this emerging opportunity
- Learn about the potential benefits, and some deployment and management solutions for software-enabled networking

Download eBook